REPORT DOCUMENTATION PAGE

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
18 June 2001	Final	12 April 1999 - 11 April 2001

4. TITLE AND SUBTITLE

Site Specific Propagation Prediction Software Tool for Communication Channel Modeling

5a. CONTRACT NUMBER

DAAD19-99-C-0026

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

George Liang

RECEIVED
JUN 2 2 2001
By

5d. PROJECT NUMBER

P-39103-EL-SB2

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Site Ware Technologies, Inc.
439 10th Street
Brooklyn, NY 11215

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U. S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)

ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES

The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by the documentation.

14. ABSTRACT

Report developed under SBIR contract. The development of a full functional deterministic propagation prediction tool that will be able to predict the propagation characteristics of radio signals in any specified outdoor and indoor physical environment will provide an important design tool for wireless communication system designers. Site Ware Technologies will develop a computer package that will be capable of characterizing the multipath propagation environment by determining the path loss, time delay profile, angle of arrival distribtution, spatial correlation and depolarization. The program will give site specific predictions in a wide range of outdoor and indoor environments for frequencies in the range of 100MHz to 10GHz. The objectives of this Phase II work is to expand on the sucessful Phase I project by implementing a number of propagation prediction features that will extend the versatility and robustness of the propagation models and encompass all the features into a sophisticated graphical user interface. The successful completion of the Phase II project will provide the U.S. Army with a comprehensive communication simulation tool as well as preparing the technologies for Phase III commercialization.

15. SUBJECT TERMS

Propagation Prediction, Radio Channel Characterization, Site Specific, Reflection, Transmission, Diffraction Scattering

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE	SAR	48	George Liang
U	U	U			**19b. TELEPHONE NUMBER** *(Include area code)* (718) 260-3412

INSTRUCTIONS FOR COMPLETING SF 298

1. REPORT DATE. Full publication date, including day, month, if available. Must cite at least the year and be Year 2000 compliant, e.g. 30-06-1998; xx-06-1998; xx-xx-1998.

2. REPORT TYPE. State the type of report, such as final, technical, interim, memorandum, master's thesis, progress, quarterly, research, special, group study, etc.

3. DATES COVERED. Indicate the time during which the work was performed and the report was written, e.g., Jun 1997 - Jun 1998; 1-10 Jun 1996; May - Nov 1998; Nov 1998.

4. TITLE. Enter title and subtitle with volume number and part number, if applicable. On classified documents, enter the title classification in parentheses.

5a. CONTRACT NUMBER. Enter all contract numbers as they appear in the report, e.g. F33615-86-C-5169.

5b. GRANT NUMBER. Enter all grant numbers as they appear in the report, e.g. AFOSR-82-1234.

5c. PROGRAM ELEMENT NUMBER. Enter all program element numbers as they appear in the report, e.g. 61101A.

5d. PROJECT NUMBER. Enter all project numbers as they appear in the report, e.g. 1F665702D1257; ILIR.

5e. TASK NUMBER. Enter all task numbers as they appear in the report, e.g. 05; RF0330201; T4112.

5f. WORK UNIT NUMBER. Enter all work unit numbers as they appear in the report, e.g. 001; AFAPL30480105.

6. AUTHOR(S). Enter name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. The form of entry is the last name, first name, middle initial, and additional qualifiers separated by commas, e.g. Smith, Richard, J, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES). Self-explanatory.

8. PERFORMING ORGANIZATION REPORT NUMBER. Enter all unique alphanumeric report numbers assigned by the performing organization, e.g. BRL-1234; AFWL-TR-85-4017-Vol-21-PT-2.

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES). Enter the name and address of the organization(s) financially responsible for and monitoring the work.

10. SPONSOR/MONITOR'S ACRONYM(S). Enter, if available, e.g. BRL, ARDEC, NADC.

11. SPONSOR/MONITOR'S REPORT NUMBER(S). Enter report number as assigned by the sponsoring/ monitoring agency, if available, e.g. BRL-TR-829; -215.

12. DISTRIBUTION/AVAILABILITY STATEMENT. Use agency-mandated availability statements to indicate the public availability or distribution limitations of the report. If additional limitations/ restrictions or special markings are indicated, follow agency authorization procedures, e.g. RD/FRD, PROPIN, ITAR, etc. Include copyright information.

13. SUPPLEMENTARY NOTES. Enter information not included elsewhere such as: prepared in cooperation with; translation of; report supersedes; old edition number, etc.

14. ABSTRACT. A brief (approximately 200 words) factual summary of the most significant information.

15. SUBJECT TERMS. Key words or phrases identifying major concepts in the report.

16. SECURITY CLASSIFICATION. Enter security classification in accordance with security classification regulations, e.g. U, C, S, etc. If this form contains classified information, stamp classification level on the top and bottom of this page.

17. LIMITATION OF ABSTRACT. This block must be completed to assign a distribution limitation to the abstract. Enter UU (Unclassified Unlimited) or SAR (Same as Report). An entry in this block is necessary if the abstract is to be limited.

Final Progress Report

To

U.S Army Research Office

For the period of:

12 April 1999 – 11 April 2001

Contract # DAAD19-99-C0026

Site Specific Propagation Prediction Software

Tool for Communication Channel Modeling

Prepared by:

George Liang

Table of Contents

TABLE OF FIGURES

Synopsis

During the two years of the SBIR grant work was focused on the design and development of the Site Specific Propagation Prediction Tool (SSPPT) that would lead to a full function radio wave propagation prediction application. The technical work was directed in two distinct areas: propagation physics and the graphical user interface. The issues that were investigated in the area of propagation physical include: propagation through vegetation, diffuse scattering from rough surfaces of the buildings, a new heuristic diffraction coefficient formulation and other algorithmic improvements. Algorithms have been implemented for modeling these phenomenons and were tested and validated with some measurements. In addition to propagation issues, a large amount of work has been undertaken in the area of the user interface. The input data such as the buildings, receivers, terrain, and vegetation can be viewed graphically in addition to some editing capabilities. It is also possible to start a simulation with the VPL engine and display a limited set of results from it. The results can be rendered within the physical environment scene or as a 2D line graph in a separate view. In addition to viewing the user interface has the capabilities of printing and saving the data and results into a single file. The culmination of the work for the two years has resulted in the release of a demonstration program that is available at www.sitewaretech.com. The demo contain some of the features described in this report and will be use to obtain feedback about its capabilities and features from interested users.

Introduction

To aid in the design of low power communication systems it is necessary to understand the physical environment in which the radio is operating. At high frequencies, (~100MHz to 10GHz) it is possible to model radio wave propagation using ray optics and the Geometrical Theory of Diffraction. Rays that represents a discrete bundle of energy are launched from the transmitter and interact with the physical environment through a number of interactions. These interactions include: specular reflections, diffraction at the edges and diffuse scattering from the surfaces due to surface roughness. The key to any robust ray tracing propagation model is to find a computationally fast way to determine the dominant ray paths, so as to provide accurate predictions.

In order to find the contributing rays in an urban environment, wherein building walls are nearly always vertical planar polygons, the Vertical Plane Launch (VPL) method has been developed. The VPL approach accounts for specular reflections from vertical surfaces and diffraction at vertical edges and approximates diffraction at horizontal edges by restricting the diffracted rays to lie in the plane of incidence, or in the plane of reflection. Compared to the full 3D shoot and bounce ray (SBR) method or the 3D image method, that can handle at most one or two diffractions for any edge orientation, the VPL approach can treat many multiple forward diffractions at horizontal edges. Unlike the vertical plane/slant plane (SP/VP) approximation, whose application is limited to low base station antennas, the VPL method can be used for rooftop antennas and areas of mixed building heights. It provides nearly identical results with the 2D method for low transmitting and receiving antennas in a tall building environment, in which case propagation takes place around buildings.

The concept of the VPL method for a rooftop antenna is indicated in Figure 1 that shows half planes originating from a vertical line through the transmitter and extending outward in one direction. As an example, the plane between the transmitter and receiver 1 in Figure 1 contains a ray that must propagate over the intervening rows of building. In

addition to the direct ray The ray reaching receiver 2 is contained in the plane shown, and is reflected from a tall building and travels over a series of lower buildings before arriving at the receiver. For receiver 3, the illuminating ray in the vertical plane undergoes diffraction at the vertical edges

Figure 1: The Vertical Plane Launch method.

of two buildings before traveling over the rooftops of lower buildings to the receiver.

It is not the purpose of this report to describe the VPL method in detail since it has been previously documented. Instead, this report is intended to describe the enhancements that have been added to the VPL method to reduce the errors in the predictions and to account for propagation effects that were not incorporated in the VPL method prior to this project. This report will also describe in detail the incorporation of the VPL method into a graphical user interface to produce a full function site-specific propagation prediction tool. The many features of the user front end will be described in text and in figures to highlight the evolving capabilities of the application.

Propagation Physics

A.1 Propagation Over Vegetation

Propagation through vegetation is a very difficult phenomenon to model due to the wide variety and random composition of vegetation that exists in a physical environment. At microwave frequencies the scattering effects of the vegetation is not negligible since the wavelength are now on the order of the dimensions of the leaves and branches. Therefore, when the propagation path must pass through a large number of tree

before arriving at the receiver it becomes necessary to account for the loss due to the diffuse scattering.

It is impractical to model the scattering from individual trees within the propagation environment primarily because there are only very weak models used to account for scattering from a single tree. Therefore, only a grove or forest of trees will be modeled. To conveniently integrate vegetation into the VPL code the area of foliage will be modeled as an arbitrarily shaped polygon similar to that of a building structure. The polygon representing the trees will contain height information representing the average height of the foliage and the material composition representing the type of vegetation. This vegetation information will be stored in a separate file and included in the VPL model as a separate layer of data. Therefore if the vegetation data exist it can be used otherwise the simulation will neglect vegetation effects. Since the format of the data for the vegetation

Figure 2: Propagation over vegetation.

area is similar to the format for the buildings it can be efficiently incorporated in the ray trace. Unlike the buildings, vertical planes that intersect with the vegetation polygons will not undergo any reflections from the faces. Instead only diffraction at the horizontal edges of the polygon will be model as shown in Figure 2. Propagation over the vegetation can be combined with other interactions with the buildings such as reflections from the building wall and diffractions from the edges.

Only the coherent part of the attenuated field will be used to model the propagation through vegetation. Unlike the incoherent field which travels through the body of the vegetation, the coherent field is the field that propagation over the canopy of the foliage through the mechanism of diffraction at the outer boundaries. There is evidence to suggest that the incoherent field has a very negligible contribution to the received field on the other side of the vegetation.

A.2 Propagation Into Vegetation

Another case to be considered is propagation into or out of the foliage area when either the transmitter or receiver is located inside or under the canopy. For this case as it is necessary to consider the loss associated with the signal penetrating through the vegetation. The propagation path will include a segment that must pass through vegetation before arriving at the receiver and is treated similar to a non-foliage path except that the height of the end point (for a receiver in foliage) is temporarily set at the height of the canopy directly above the actual receiver. The ray path may undergo any number of building interactions including reflections and edge diffractions before arriving at the temporarily elevated receiver point. The path loss for the propagation path is then calculated at this point. An additional loss representing the diffuse scattering down to the actual receiver (presently preset at 20dB) is added to this path loss to arrive at the total path loss for the propagation path.

In order for reciprocity to hold the same technique is employed if the transmitter is located inside or under the foliage area. The propagation path starts at the height of the canopy directly above the transmitter and propagates around the physical environment before arriving at the receiver. The path loss is calculated based on this elevated start (transmitter) point and an additional loss of 20dB is added to account for the propagation through the canopy from the actual transmitter.

B. Diffuse Scattering from the Buildings

To simulate diffuse scattering from the building walls it is necessary to model small sections of a wall initially as receivers and then treat them as secondary (scattering) sources. Figure 3 shows a simple example of diffuse scattering from the walls of a building while neglecting other propagation phenomenon such as specular reflections and diffractions. In Figure 3 the real transmitter illuminates virtual receivers located at the center of a predefined section of the building wall and the virtual receivers that are visible to it illuminate the real receivers. The first step is to construct a list of virtual receivers and transmitters on the building walls. This is done by subdividing each wall of every building into a predetermined area and then placing a virtual receiver at the center. Even

9

if an entire wall of a building is smaller than the predefined area each wall must have a minimum of one virtual receiver. The next step is to determine the set of virtual receivers that are visible the real receivers. Each real receiver is tested against all the virtual receivers to determine if there is a line-of-sight path between them that is within a certain distance. The result is that for each real receiver a list of visible virtual receivers are found. A union of the lists of visible virtual receivers is created to determine

Figure 3: Diffuse scattering from building walls.

a single list of virtual receivers in the scene where only one unique virtual receiver exists for a subsection of a wall. In order for reciprocity to hold it is necessary to determine the diffuse scattering from the transmitter (source) end of the link. The same procedure used to find the virtual receiver also applies to finding the virtual transmitters but since there are only a few real transmitters in the scene the number of virtual transmitters is small.

After all the virtual receivers and transmitter are found, a simulation is started with the VPL engine. The ray trace will launch vertical planes in all directions around a 360° circle in the horizontal plane and undergo the user-defined number of specular reflections, vertical diffractions as well as diffractions at horizontal edges. Vertical planes that illuminate real receivers are stored as rays that have undergone a combination of the reflections and diffractions only. Vertical planes that illuminate virtual receivers are stored as source rays for subsequent calculations to determine the diffuse scattering from the receiver end of the link. Finally vertical planes that illuminate virtual transmitters are stored as source rays for subsequent calculations to find the diffuse scattering at the transmitter end of the link.

To calculate the diffuse scattered power between a virtual and real receiver no other ray tracing is required since the power of the illuminating rays and the list of real receivers is known for any given virtual receiver. Therefore for each illuminating ray the scattered power due to the surface roughness at each visible real receiver given by,

$$P_{rec} = K(\hat{n}_{VRx} \bullet \hat{n}_{Rx})(\hat{n}_{VRx} \bullet \hat{n}_{ray})P_{ray} \qquad (1)$$

Where $\hat{n}_{VRx}$, $\hat{n}_{Rx}$ and $\hat{n}_{ray}$ are the outward unit normal vector of the virtual receiver, unit vector of the segment joining the real and virtual receiver and the unit vector of the illuminating ray respectively. The variable P_{ray} is the power in Watts of the illuminating ray at the virtual receiver and K is the diffuse scattering coefficient and is given by,

$$K = \frac{C(Area)}{2\pi R^2} \qquad (2)$$

The *Area* is the effective area of the scattering object and R is the distance between the scattering center and the observation point. The variable C is a constant that determines the fraction of the incident power that is scattered. The determine an appropriate value for C the total scatter power density must be integrated over a unit sphere and is given by,

$$P_{scattered}^{Total} = \int_0^{\pi/2} \int_0^{2\pi} C \frac{P_r \cos\theta_s}{2\pi R^2} R^2 \sin\theta_s \, d\phi_s \, d\theta_s = \frac{CP_r}{2} \qquad (3)$$

Where P_r is the incident or illuminated power. Then the fraction of the scattered power is,

$$\frac{P_{scattered}^{Total}}{P_r} = \frac{C}{2} \qquad (4)$$

Therefore if 10% of the incident power is scattered due to the rough surface then the constant C is equal to 0.2.

To determine the scattered power at the transmitter end of the link additional ray traces must be performed. Each virtual transmitter is treated as a source point and a ray trace is conducted from each one. Vertical planes launched from the virtual transmitters

can undergo the same number of reflections and diffractions as planes from the real transmitter. Vertical planes from the virtual transmitters can only illuminate real receivers and are treated as rays that are scattered from the rough surfaces near the transmitter. Therefore, to find the scattered power at a real receiver due to the rough surface scattering from surfaces near the real transmitter every combination of the illuminating ray at a real receiver and source rays at the virtual transmitter must be calculated. The scattered power for one combination is given by,

$$P_{rec} = K(\hat{n}_{VTx} \bullet \hat{n}_{Tx})(\hat{n}_{VTx} \bullet \hat{n}_{ray})P_{ray} \tag{5}$$

Where $\hat{n}_{VTx}$, $\hat{n}_{Tx}$ and $\hat{n}_{ray}$ are the outward unit normal vector of the virtual transmitter, the unit vector of the segment join the real and virtual transmitters and the unit vector of the departing ray from the virtual transmitter respectively. The value of K is given by Equation 2 and P_{ray} is the power of the illuminating ray at the virtual transmitter.

By including diffuse scattering from the wall of the buildings it is hoped that it would increase the signal strength in region where the predictions are lower than the corresponding measurements thereby lowering the standard deviation. This is especially true for the region in the deep shadows where it is necessary to use two vertical edge diffractions to reduce the error. Because rough surface scattering is a lower order effect and requires more computation resources to model relative to modeling specular reflections and diffractions it become very important to determine the importance of diffuse scattering to the overall accuracy of the model. Figure 4 shows a preliminary test case

Figure 4: Comparison between prediction with and without diffuse scattering.

comparing the predictions with and without diffuse scattering with the measurements. This case is for TX1b a street level transmitter (10 meters above ground) located in Rosslyn, Virginia. The measurements are indicated by the green line while the predictions with no diffuse scattering is the black curve and the predictions with diffuse scattering for two different values of the constant C is shown by the blue and purple curves. Figure 4 shows that including rough surface scattering from the building walls produces some improvement in the predictions especially in shadowed regions where the differences where previously greater than 20 dB. On the other hand, the diffuse scattering has negligible contribution in the LOS and near LOS region since higher order propagation phenomenon will dominate.

The algorithms and techniques used to model rough surface scattering still require additional testing and refinement. Parameters such as the size of the subsection of the walls and the maximum distance between a virtual receiver or transmitter to a real receiver and transmitter should be varied to determine their effects on the accuracy of the predictions. The example shown in Figure 4 uses a 15-meter sub sectioning area for the walls and a 50-meter maximum distance between the virtual and real receivers/transmitters. Decreasing the subdivided area on the wall and increasing the maximum distance between the virtual and real receivers/transmitters will increase the number of virtual receivers/transmitters and result in the increase in computation time.

C. Dielectric Wedge Diffraction Coefficient

The VPL engine has been using a heuristic UTD diffraction formulation developed by Luebbers to account for the excess loss due to diffraction at the building edges. The advantage of this formulation over the formulation for the diffraction coefficient of an absorbing wedge is that it will correctly account for the transition region around the reflection shadow boundary. By taking into account the reflection shadow boundary, rays that would have been considered too weak to have been kept with the absorbing wedge formulation will now be retained and contribute to the overall signal

strength. This enables better predictions in the shadow regions where the predictions have historically been more pessimistic when compared to the measurements.

The main drawback of Luebbers' formulation is that in the deep shadow regions that calculated field is lower than the calculated field for the absorbing wedge. The diffracted field also contains some deep nulls due to zeros that are the result of cancellation of the significant terms in the diffraction coefficient. These nulls cannot be explained by the physics of the problem but is merely due to the mathematical formulation. Therefore it is necessary to develop a new diffraction coefficient that will be able to better model the diffraction phenomenon at the building edge.

C.1 Formulation

The new heuristic UTD diffraction coefficient builds on the formulation for the diffraction coefficient for an absorbing wedge by adding the terms that represent the reflection shadow boundaries. The diffracted electric field for a single diffraction at dielectric wedge with an arbitrary wedge angle is given by:

$$E_{diff} = E_o \frac{e^{-j(k(\rho+\rho')+\frac{\pi}{4})}}{\sqrt{\rho\rho'(\rho+\rho')}} D(\phi,\phi') \tag{6}$$

where E_o is the incident field at the edge. The terms ρ and ρ' represent the distance between the source to the edge and between the edge to the observation point respectively. The term $D(\phi,\phi')$ is the diffraction coefficient and is given by:

$$D(\phi,\phi') = -\frac{1}{\sqrt{2\pi k}}\left(\frac{1}{\pi-|\phi-\phi'|}F(kLa^-(\phi-\phi'))+\frac{1}{\pi+|\phi-\phi'|}F(kLa^+(\phi-\phi'))\right) \tag{7}$$

or by:

$$D(\phi,\phi') = -\frac{1}{\sqrt{2\pi k}}\left(\begin{array}{c}\frac{1}{\pi-|\phi-\phi'|}F(kLa^-(\phi-\phi'))+\frac{1}{\pi+|\phi-\phi'|}F(kLa^+(\phi-\phi'))\\ +\Gamma_o\frac{1}{\pi-(\phi+\phi')}F(kLa^-(\phi+\phi'))+\Gamma_n\frac{1}{(\phi+\phi')-(2n\pi-\pi)}f(kLa^+(\phi+\phi'))\end{array}\right) \tag{8}$$

14

whichever is greater in magnitude. The angles ϕ' and ϕ are the incident and diffracted

ray angles with respect to the illuminated face of the wedge and if ψ is the wedge angle

then n is related to ψ by:

$$\psi = (2-n)\pi \tag{9}$$

Equation 7 is just the diffraction coefficient for an absorbing wedge and includes only the

shadow boundary due to the incident field. The two additional terms at the end of

Equation 8 represent the shadow boundary due to the reflected field. Since the

diffraction coefficient is a selection of either Equation 7 or 8 the minimum value of

$D(\phi,\phi')$ will be the value given by Equation 7. Depending on the polarization, the sign

of Γ_o and Γ_n can be negative which can reduce the magnitude of Equation 8 to less than

Equation 7. The Γ_o and Γ_n terms are the reflection coefficient for the incident and

diffracted rays respectively and is given by:

$$\Gamma_{\parallel} = \frac{\cos\theta_i - \sqrt{\varepsilon_r}\cos\theta_t}{\cos\theta_i + \sqrt{\varepsilon_r}\cos\theta_t} \tag{10}$$

for parallel polarization and

$$\Gamma_{\perp} = \frac{\sqrt{\varepsilon_r}\cos\theta_i - \cos\theta_t}{\sqrt{\varepsilon_r}\cos\theta_i + \cos\theta_t} \tag{11}$$

for perpendicular polarized electric field. Where θ_i and θ_t are the incident and

transmitted angles with respect to the normal of the wall surface. The transmitted angle

θ_t is related to the incident angle by Snell's law and is given by:

$$\theta_t = \sin^{-1}\left(\frac{1}{\sqrt{\varepsilon_r}}\sin\theta_i\right) \tag{12}$$

The incident angle with respect to the illuminated face Γ_o is related to the angle of the

incident ray by:

$$\theta_0^i = \left|\phi' - \frac{\pi}{2}\right| \tag{13}$$

The incident angle with respect to the "n" or un-illuminated face is also related to

the incident ray angle:

15

$$\theta_n^i = \left| \frac{\pi}{2} + n\pi - \phi \right| \tag{14}$$

provided that $\left| \frac{\pi}{2} + n\pi - \phi \right| < \frac{\pi}{2}$ otherwise θ_n^i is equal to:

$$\theta_n^i = \left| \frac{\pi}{2} + n\pi - \phi \right| - m\frac{\pi}{2} \tag{15}$$

such that $\left| \theta_n^i \right| < \frac{\pi}{2}$. The function $F(kLa^\pm(\phi \pm \phi'))$ is the complementary error function that corrects the poles associated with the singularity of the shadow boundaries.

C.2 Comparison

Figure 5 illustrates the difference between the diffraction loss formulation of the absorbing, Luebbers dielectric and the new heuristic diffraction coefficient for a right angle (90°) wedge with a parallel polarized incident rays at 10°. A dielectric constant of $\varepsilon_r = 6$ was used for the material characteristic of the wedge. This case represents the scenario of a wave diffracting at a vertical edge of a building. The plots of Figure 5 show the excess loss at a fixed distance (50m)

Figure 5: Comparison of the diffraction coefficient for a 90-degree dielectric wedge.

around the diffracting edge where 0° starts at the illuminated wall and 270° is at the shadowed wall. Figure 5 clearly shows that Luebbers' formulation immediately diverges from the absorbing wedge calculation as it enters the geometric shadowed regions. In the deep shadows, >250° the difference in excess loss varies from 3.5 to over 9.0dB. This difference is significant in the deeply shadow region since the signal that arrives in this region is usually heavily attenuated. The new heuristic formulation would improve the predictions is the deep shadows where it has traditionally been more pessimistic when compared to the measurements.

Luebbers' formulation also shows a deep null at 75° in Figure 5 and can not be readily explained since it does not fall near any of the shadow boundaries. In general, the excess loss for the backward diffracted field (0° – 90°) is much greater than for the absorbing wedge.

Figure 6 shows a 120° vertically polarized incident field illuminating a 90° dielectric wedge for the different diffraction coefficients formulations. For this case, there is no incident shadow boundary as indicated by the plot of the absorbing wedge diffraction. Instead there are two reflection shadow boundaries at each of the two faces that make up the corner. Figure 6 clearly indicates that Luebbers' formulation calculates a smaller excess loss in the region between the two shadow boundaries. But for the regions near the surfaces of the walls Luebbers' coefficient exhibits deep nulls that

Figure 6: Comparison of the diffraction coefficient for a 90-degree dielectric wedge.

are physically unexplainable. Therefore with the new heuristic UTD formulation, Figure 6 shows that forcing the minimum value to be the absorbing wedge value can eliminate the deep null in Luebbers formulation.

The overall result of the new UTD diffraction coefficient formulation is that the total received signal in the shadowed region will be larger. This will enable the predictions to have closer agreement with the measurements.

D. Multiple Diffraction Over Buildings

The current method of calculating diffraction across multiple edges is to cascade the UTD diffraction coefficient up to four edges. Although this method is computationally efficient it does not adequately describe the propagation phenomenon especially when there are many edges and the difference in heights between the edges are small. The error becomes the greatest when all the edges are at the same height. To calculate the field over N edges of equal heights using the UTD diffraction coefficient the field is in the order of:

$$\overline{E} \approx \left(\frac{1}{2}\right)^N \tag{16}$$

On the other hand, using numerical integration techniques the field is closer to the approximated value of:

$$\overline{E} \approx \frac{1}{N+1} \tag{17}$$

Therefore when the propagation path must diffract over many buildings to reach the receiver it is necessary to improve the method for determining the field.

One computationally efficient method is to approximate the diffraction of the edges that fall within the Fresnel zone by merely counting the number of edges that fall between isolated diffraction edges. An isolated diffraction edge is an edge that is greater (taller) than a specified Fresnel zone and thus the signal must diffracted from this edge.

Figure 7: Diffraction over multiple horizontal edges

Non-isolated diffraction edges are edges that contribute to the diffracted field because they block part of the energy contained in the Fresnel zone. Figure 7 shows an

18

example of the vertical profile of the horizontal diffracting edges for a ray path. It contains a couple of isolated diffracting edges and a number of other edges that partially obstruct the Fresnel zone.

To find the all the isolated diffracting edge it is necessary to first construct a line between the transmitter and the receiver and then find the height value h given by:

$$z_n = z_{tx} + d_n \frac{(z_{rx} - z_{tx})}{d_{rx}} \tag{18}$$

on this line at the position of an edge (d_n). To determine if the height of the edge is greater than the Fresnel width at the position of the edge the following condition must be met:

$$F_n = \frac{(h_n - z_n)^2}{\lambda} \frac{d_{rx}}{d_n(d_{rx} - d_n)} > 0.36 \tag{19}$$

If the condition in Equation 19 is true then the edge is a potential isolated diffraction edge. All the edges between the transmitter and the receiver are tested to find the largest value of F_n. The edge that has the largest value for F_n is marked as an isolated diffraction edge. The horizontal edges are now bisected by the isolated diffraction edge into two segments, the segment between the transmitter and the isolated diffraction edge and between the edge and the receiver. For each of these two segments, the same test is performed recursively until all intermediate edges satisfy the condition of $F_n < 0.36$.

Subsequent to finding all the isolated diffracting edges it is necessary to test the remaining edges to determine if they partially obstruct the Fresnel zones. Between each segment (transmitter-edge, edge-edge and edge-receiver) all the edges that have a height greater than or equal to:

$$h_n \geq z_n - W_e \tag{20}$$

is considered to by a partially obstructing edge. The term W_e is an effective Fresnel width for the lower half of the Fresnel zone and is equal to:

$$W_e = \frac{1}{3.5} \sqrt{\lambda d} \tag{21}$$

where d is the average separation distance between the edges. When all the Fresnel edges have been identified, they are grouped according to the segment of the propagation path that they are located in. For example shown in Figure 7 there are two edges in the segment between the transmitter and first isolated diffracting edge and between the first and second isolated diffracting edge. However four edges fall in the segment between the second isolated diffracting edge and the receiver.

The total path gain for the example shown in Figure 7 is:

$$PG = \left(\frac{\lambda}{4\pi}\right)^2 \prod_{p=1}^{2} \frac{D^2(\phi,\phi')_p}{\sqrt{d_3}} \prod_{p=0}^{2} \frac{1}{(d_{p+1}-d_p)} \prod_{p=1}^{3} \frac{1}{\left(N_p^e + 1\right)^2} \tag{22}$$

where $D(\phi,\phi')_p$ is the diffraction coefficient for the pth isolated diffracting edge and d_p is the distance of the pth edge from the transmitter. The first three terms in Equation 22 are just the UTD path gain when only the isolated diffracting edges are considered. The last term is the additional loss associated with the Fresnel edges where N_p^e is the number of effective edges between the pth and the $p - 1$ edge.

E. Propagation Over Very Large Areas

Problems arise when a ray must propagate over very long distances due to large separation between the transmitter and the receiver or because the physical environment covers a very large area. The two problems can be summarized as follows:

1. Too many reflection branches due to the ray passing over a large number of buildings.
2. Over sampling of near buildings and under sampling of far buildings.

The first problem stems from the fact that a reflected ray can potentially be generated at every wall that faces the incident ray. Therefore when the incident ray must pass over a large number of buildings before reaching the outer limits of the building database it results in an unmanageable number of rays being generated. The number of rays (*NoR*) that could be generated by a single incident ray from the transmitter is given by:

$$NoR = N^n \tag{23}$$

20

where n is the maximum number of reflections and N is the average number of buildings that a ray will pass over. Therefore, the number of rays that can be generated is exponential with the number of buildings that it passes over as the base. It becomes quite evident that the number of rays will quickly become unbounded as N increases for even a few reflections.

The second problem of over sampling the buildings near the transmitter is illustrated by Figure 8 that shows that at a given angl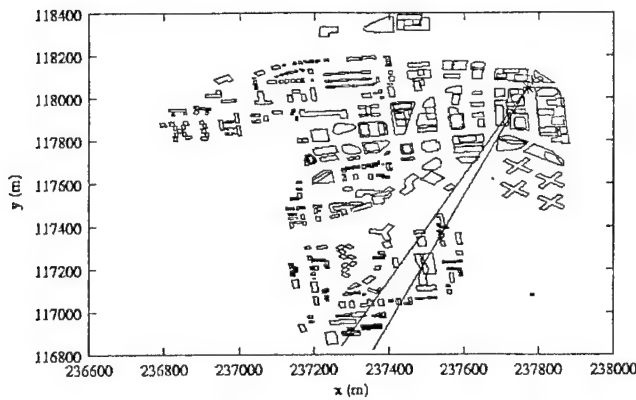e interval the two adjacent rays will completely miss the filled in building located at the bottom edge of the database. In order to sample (allow for a reflection at) the filled in building it would be necessary to reduce the angle interval between two adjacent rays from the transmitter. The immediate result of a reduce angle interval is to increases the

Figure 8: Near/Far problem, over sampling near the transmitter and under sampling far from it.

number of rays that are launched from the transmitter. The increased number of rays from the transmitter will produce an exponential number of reflected rays given by Equation 23. A secondary result is that the buildings near the transmitter are over sampled, meaning that many more rays are reflected from its walls than is needed for an accurate prediction. The problem of over sampling can be characterized as a near/far problem.

The solution to the problem of too many reflection branch rays being generated is to create bounding area around the transmitter and receivers. Figure 9 shows a scenario where the transmitter and receivers are located at some distance apart and it is necessary for the propagation paths to travel over an intervening distance that contains no receivers. A bounding area is defined around the group of receivers by determining the maximum and minimum extent of the receivers and then adding an additional fix border to increase

the total area of the bounding box. All the buildings (vertices) that fall inside this bounding area are marked as buildings that reflection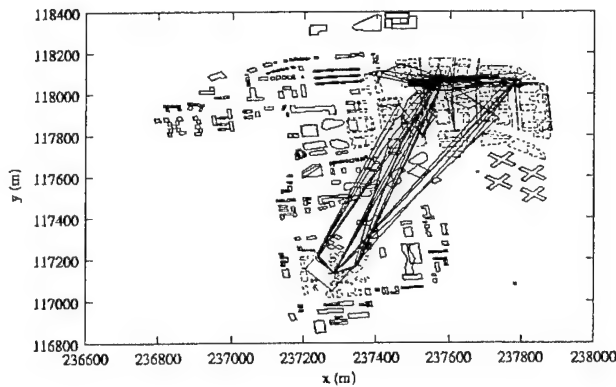s from the face of the wall and diffractions from its vertical edges can occur and is shown as a dotted outline in Figure 9. Conversely, buildings located outside of the bounding area are marked so as to indicate that no reflection or vertical edge diffractions is allowed and is shown as a solid outline in Figure 9. However, diffraction at a horizontal edge or propagation over the building is allowed for all buildings. Similarly, this procedure is used to find all the buildings that reflections or vertical edge diffraction can occur at for a predefined radius around the transmitter.

Figure 9: Building bounding areas for long propagation paths.

The effect of these bounding areas around the transmitter and receivers is to limit the generation of reflection branch rays and secondary diffracted rays from vertical edges when the incident ray travels outside of these bounding areas. It is evident that it is important to consider the effects of the intervening buildings especially if it will obstruct the propagation path. On the other hand, although reflections and vertical edge diffractions will occur at those buildings outside the bounding area it will not constitute the important contributions at the receiver. In short, the most important contributions to the total signal at the receivers are comprised of the propagation paths that have undergone local scattering around the transmitter and receiver.

As mentioned early, an obvious (and simple) solution to the near/far problem is to reduce the angle interval between two adjacent rays until the arc length at the maximum extent is small enough to sample even the smallest wall. The significant drawback to this solution is that too many rays are launched from the transmitter and will increase the execution time of a simulation. In order to limit the number of rays and reduce the arc-

length between adjacent rays, a ray insertion technique was implemented. Figure 10 illustrates the ray insertion concept as it is applied to the VPL ray launching technique. The thick rays originate from the transmitter or a diffracting corner of the building. In the horizontal plane these rays can undergo reflections from building wall although it is not shown in Figure 10 for the sake of simplicity. At some distance r the arc-length between the two adjacent rays from the transmitter is approximately equal to w. We want to find r so that the value of w is always less than the average width of the building walls.
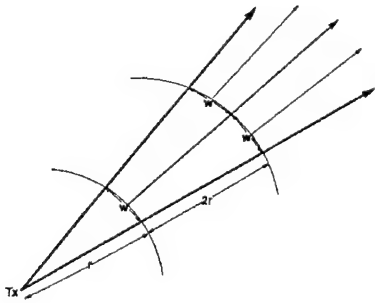

Figure 10: Ray insertion technique.

Beyond r the arc-length is greater than w, therefore we insert a ray halfway along the arc between the two adjacent rays and trace it forward. At $2r$ the arc-length between the inserted ray and either one of the source rays is equal to w again, which requires us to insert a ray on either side of the first inserted ray and is traced forward. This process is continued until the entire length of the source ray has been covered. The technique of ray insertion results in a sparse number of rays near the transmitter while maintaining a constant maximum interval width at large distances from the transmitter.

Graphical User Interface

The graphical user interface is intended to aid in the visualization and manipulation of the data input and the output from the prediction engine. The integration of the propagation prediction engine into the application allows the user to conduct propagation studies of various physical environments in a seamless fashion. A virtual scene of the physical environment can be constructed by importing the buildings, terrain and foliage from different formats into the application. Each layer (terrain, buildings, foliage) can come from its own separate source and combined to make the city model. The system is designed so that adding support for more file formats, or even custom file formats for specific customers, can be done easily in a separate DLL from the rest of the program. For example, DEMs and DTEDs are already supported, and support for Shapefiles, VMap, and DXF formats are coming soon. In addition, the program will be able to combine multiple files into a single layer, piecing together many patches of terrain, for instance, or inserting many buildings individually. Within this scene a set of receivers and at least one transmitter is necessary since the simulation of the radio wave propagation is done from point to point. The application provides dialog pages to guide the user through the procedure for entering the correct parameters need to obtain accurate simulation results. As the engine is performing its calculation the user is able to graph or view the intermediate results in the scene. This enables the user to get an immediate sense of the propagation characteristic of radio wave in the physical environment of interest. Upon the completion of the simulation the user can plot the final results, print a hardcopy and save the entire scene and results for analysis at a later time.

The graphical user interface is currently implement with Microsoft Visual C++ version 6.0 with SGI OpenGL version 1.1 graphics library. The application is currently ported to run on Microsoft Windows (NT and 98) machines only. The virtual scene is completely rendered by pumping all the data through the OpenGL pipe. This means that when the databases become very large the latency of the scene also becomes very large. An example is shown in Figure 5 where the building and terrain data is render for an area that is approximately 81 square kilometers. The resolution of the terrain data is 10 meters

and there are over 32,000 buildings in the scene. To render the scene shown in Figure 5 in real-time is not currently possible. It will be necessary to develop more efficient algorithms, such as Level of Detail in order to be able to view large areas practically. On the other hand the graph view uses Microsoft's built-in Graphics Device Interface (GDI) instead of OpenGL to render the objects in the graph view.
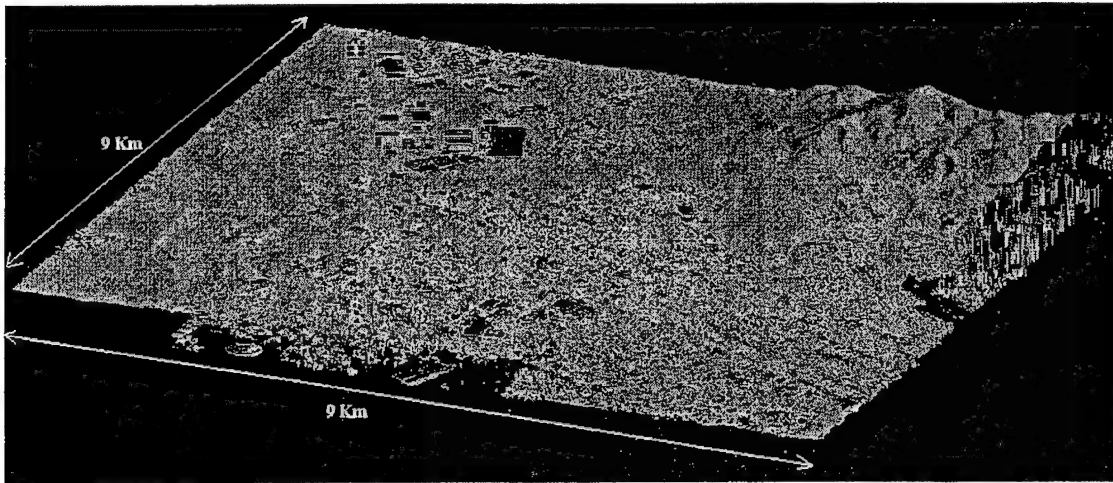

Figure 11: Viewing a very large area.

The VPL engine is written in ANSI compliant C/C++ and unlike the user interface can be ported almost any computer platform that has a C/C++ compiler. The VPL engine can be started with two different methods, locally or remotely. When the engine is started locally it is launched as a thread in the application on the same computer that is hosting the user interface. Starting the engine as a thread it allows the user to continue to perform other operations in the application. This is especially important since a single simulation can take many hours to complete. In order to add some flexibility to the application the engine can be launched on a remote computer. To enable the computation engine to run on a platform that is separate from the platform hosting the user interface, Remote Procedure Calling (RPC) is used. The use of RPC allows the application to utilize super or massive parallel computers to perform the simulation and then send the results back to the user interface.

A. Viewing the Buildings

Buildings are currently loaded from the formatted ASCII file that specifies the location of the vertices of the roof (x,y,z), the height of the building and the material characteristics of the wall. Figure 12 shows the rendering of the buildings in perspective view. The walls of the building are always vertical while the orientation of the roof is dependent on the height value of the vertex. The buildings can also be viewed from the top view only. Convex and non-convex polygons can be used to represent the roof. Roofs are rendered with a wire-frame edge so that it is possible to distinguish different roofs that overlap other
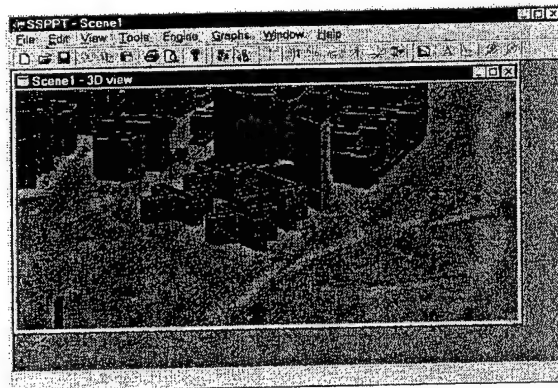

Figure 12: Viewing the buildings.

roofs in the top view. The walls of the buildings are also textured with the following materials: glass, metal, wood, stone, concrete, and brick. Different bitmaps are used for each material to provide a somewhat realistic rendering of the building. Texture mapping is implemented so that the texture element (i.e. brick) always has the same absolute size so that when the view is zoomed in the details become more pronounced. The propagation prediction engine uses the material characteristic of the wall to specify the dielectric constant when calculating the reflection coefficient.

The mouse can select individual buildings while groups of buildings can be selected by holding down the Ctrl key while selecting with the mouse. Building(s)can be selected in the 3D or 2D view as well as from the building list. User can move selected buildings in the X-Y plane only on the top view and in Z direction only in the perspective view. Selected building(s) can also be deleted. All the move and delete operations can be undone/redone.

The interior structure of the building can also be loaded from a formatted ASCII file. The indoor information only provides a general description of the building's interior and contains:

- Average height of the ground floor
- Average height of the other floors
- Average room width
- The type of material for the interior/exterior walls
- The number of elevator shafts and the coordinates of the vertices that make up the polygon that describes the elevator shaft.

The interior information is used when propagation prediction into the building is needed. The indoor database is currently on graphically shown in the building view but rather it is tabulated in the building list view.

SSPPT's main window is a 3-dimensional view of your city model. After each file loads, you will be able to see your updated city model on the screen. This allows the user to take either a birds-eye view, the view from street level, or anywhere in between as he or she desires. Being able to see the view from any angle allows better understanding of the nature of the problem, easy identification of the major obstacles to propagation, and more flexibility in examining the data. The model is initially shown to you from above, but can be viewed in one of two modes: Top View and Perspective View. Figure 13 shows the toolbar buttons that are used

Figure 13: Scene view buttons.

to switch between the two different views. Pressing the left button (shown depressed) will put you in top view mode; the right button will put you in perspective mode. The view starts out in top view mode, looking at the model from directly overhead. In perspective view you can see the city from any angle. Moving around the scene is done easily with the mouse. Dragging with the left button slides you around the scene, the right button rotates the scene, and the mouse wheel zooms you in and out.

B. Displaying the Terrain

The terrain data is currently loaded from an ASCII file containing the tabulated X, Y and Z coordinates of the terrain points. The terrain is represented as a set of tesse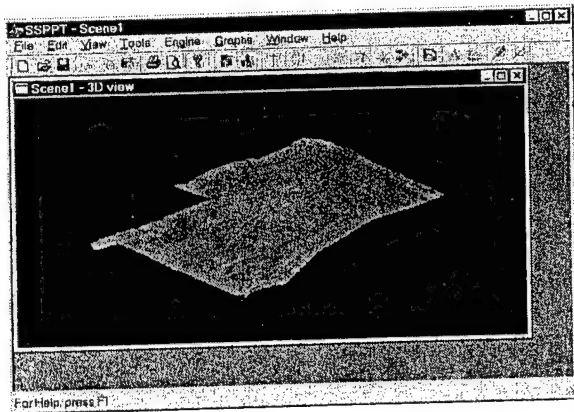llating triangles that covers the area. Shading is used to render a more realistic terrain and to provide a visual aid to identifying the terrain features. The outward normal for each triangle is calculated and when combined with a virtual light source enables shadows to be generated. The resolution of the terrain data is specified in the file and can be any real value. The tradeoffs for viewing a very large area high-resolution terrain data is that it can very easily overwhelm the physical system memory, which leads to swapping and potentially to thrashing.

Figure 14: Viewing the terrain.

The terrain can be viewed in 3D perspective and 2D top views. The views can be zoomed in or out, rotated or translated. There is currently no capability to edit the terrain in the view.

The ability to import terrain data from other formats such as: DEM, DTED and others, will be implemented in future versions of the application so that a larger set of data is available for the application.

C. Displaying and Creating Receivers

The receivers are currently loaded from an ASCII file or can be created interactively. The following information is stored for each receiver: receiver ID, X, Y and Z coordinates and height. When the receivers are loaded from a file or created interactively the height of the receivers above the ground is entered by the user. This specified height applies to all the receivers in the scene. Figure 15 shows a receiver

being rendered in the scene and is represented by a cube situated on top of a square peg.

At very large distances the receivers are displayed as a minimum of two pixels so that they do not become indistinguishable. Therefore no matter how far the user zooms away from the receiver there will always be a red dot for the receiver.
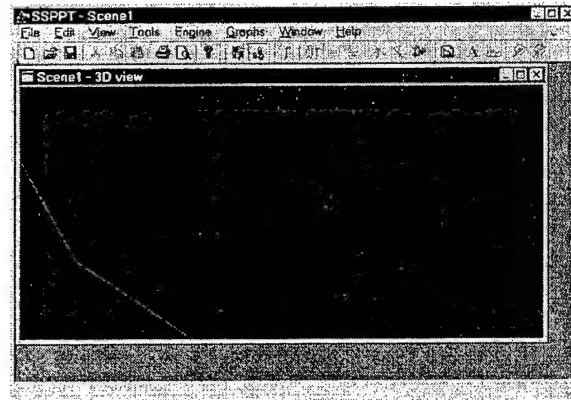
Individual receivers can be selected by the mouse pointer with the selected receiver changing colors from red

Figure 15: Rendering a receiver point.

to yellow. Multiple receivers can be selected by holding depressing the **Ctrl** key while selecting with the mouse. Receiver(s) can also be selected from the receiver list view. When receiver(s) are selected the corresponding line in the list view is highlighted in addition to the change in color in the scene. Selected receiver(s) can by moved horizontally in the 2D top view and vertically in the 3D perspective view. Selected receiver(s) can also be deleted. All move and delete operations can be undone/redone.

While the data for the buildings and terrain can come from a number of different sources, it is usually not possible to have a predetermined set of receiver data. Therefore it become necessary to create the receivers once the building and terrain scene is set in the view. There are three different methods of receiver placement:

1. Placing a single receiver positioning the mouse in the 2D top view.
2. Placing a strip or receivers by specifying the path in the 2D top view and the distance between the receivers.
3. Placing an area of receivers by specifying the rectangular area in the 2D top view to be covered by the mesh of receivers with specified distance between them.

A user can select which of the three method of receiver placement by pressing the
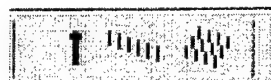

Figure 16: Receiver placement selection buttons.

appropriate button on the receiver placement toolbar shown in Figure 16. The leftmost button places a single receiver with each mouse click, the middle button places a row of receivers and the right button places receivers within an area.

For all three methods of receiver placement the receiver will be properly placed on the top of the terrain or the roof of the building. It is also possible for the user to select an option so that the receivers that have a building beneath them will not be created. If there is no building or ground under the position where a receiver is to be created then it will be created with an elevation at the bottom of bounding box that defines the terrain/receivers/buildings scene. Figure 18 shows an example of receivers place into a specified area with equal separation between each receiver.

Figure 17: Receiver placement over an area.

It also shows that when a receiver has a building under it program will automatically place the receiver at the specified height above the rooftop. Figure 17 shows an example of receivers placed along a piecewise
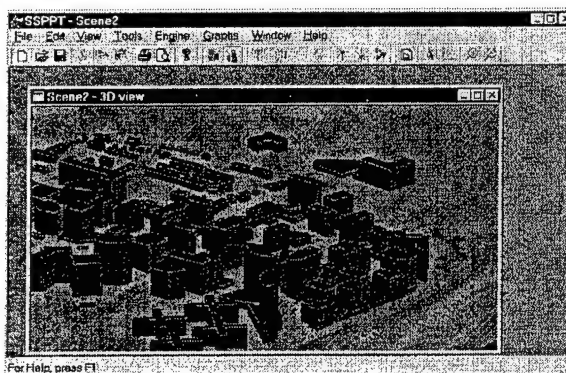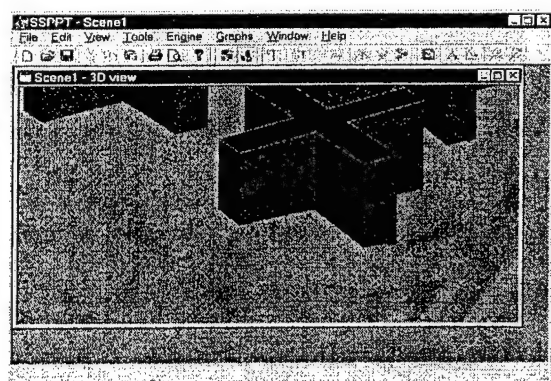

Figure 18: Receiver placement along a line.

line at the specified interval. It is possible to specify as many segments as needed. For both the line and area receiver placement a receiver spacing dialog shown in Figure 20 will appear to allow the user to input the desired receiver spacing and to select the option to place the receiver on the rooftop.

When a user double-clicks on the receiver in the 3D View or in the Receivers List view with the mouse cursor, a receiver property dialog similar to the one shown in Figure 19 will appear. The receiver properties dialog displays the receiver's Id, receiver's height above the ground, and its X, Y, Z coordinates.
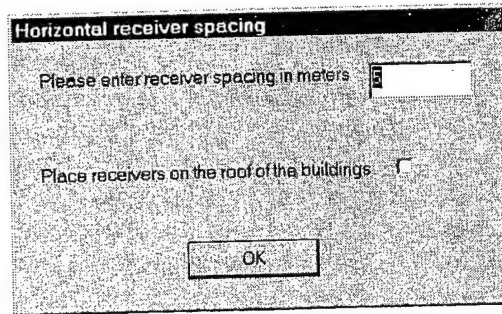

Figure 20: Dialog for receiver placement.

The dialog will also list all the rays that have illuminated the receiver for a specific combination of transmitter height and frequency. The total length of the ray path in meters, the calculated power in dB, the angle that the ray left the transmitter in degrees reference from the positive x-axis in the counter-clockwise direction, and the ray's arrival angle at the receiver in azimuth and elevation are list for each ray. Below the list of individual rays in a separate field
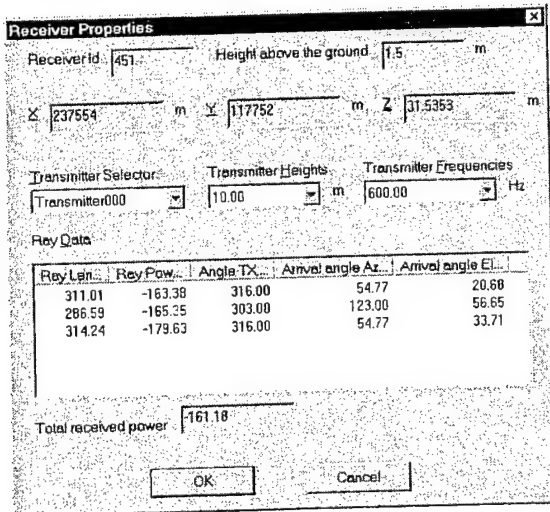

Figure 19: Receiver properties dialog.

the total power in dB is calculated and displayed. The receiver property dialog allows the user to view and information of each individual ray path that arrives at any given receiver point.

D. Creating and Displaying the Transmitter

The transmitter(s), like the receivers are normally a user placed object that must be created interactively. Each transmitter is created individually by pressing the transmitter placement button [T] and positioning the mouse cursor in the scene. Transmitter can only be created in the 2D top view. The logic that is used for the

31

placement of the transmitter is similar to that used for the receivers. If there is a building under the desired transmitter location then the base of the transmitter will be position on the rooftop. Otherwise the transmitter is place on the surface of the terrain. Transmitters
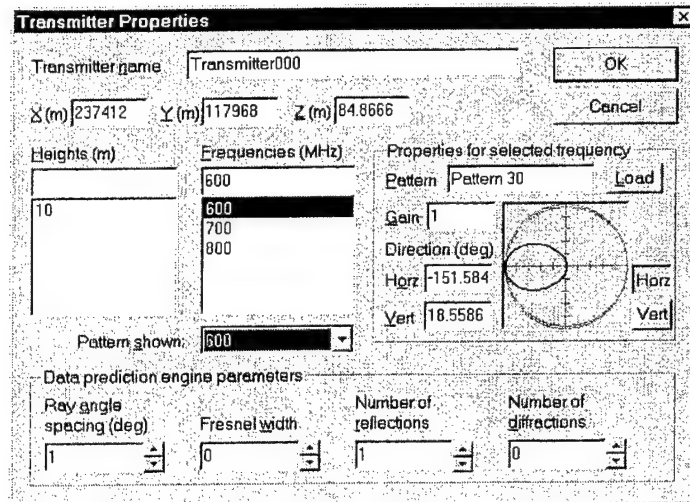

Figure 21: Transmitter properties dialog.

are rendered as a vertical pole with a cross at each specified heights. No matter how far the user zooms away from the transmitter it will always appear as a green dot to indicate its location.

When the location for the transmitter is selected a transmitter properties dialog shown in Figure 21 will appear. Creating or double clicking on a transmitter in the scene or the transmitter list view will pop up the transmitter properties dialog displaying the following information:

- The transmitter name.

- The X, Y, Z coordinates of the transmitter.

- The heights above its base at which the transmitter emits its signal.

- The frequencies at which it operates.

For a selected frequency, the antenna's radiation pattern can be specified by the user and loaded from a file. Additional antenna parameters such as:

- The gain of the antenna.

- The angle of the antenna bore sight in azimuth with respect to due east moving in a counter clockwise direction.

- The angle of the antenna bore sight in elevation relative level ground.

- The antenna pattern: the relative strength of the signal for each degree relative to the bore sight display in a polar plot.

can also be entered in the respective field of the dialog. If a frequency is specified in the
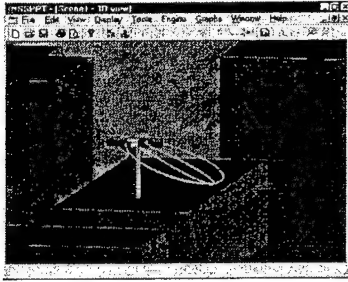

Figure 22: Transmitter with the radiation pattern displayed.

pattern shown field, the appropriate radiation pattern will be display in the city model window as shown in Figure 22. The radiation pattern is display as an outline of the pattern in azimuth and elevation plane. The user can use the mouse cursor to select the radiation pattern and change the direction the bore sight pointing direction. The same radiation pattern is used at each height of a transmitter with multiple heights but the pattern is shown on the highest height.

The transmitter properties dialog also contains controls that allow the user to specify the parameters that the propagation prediction engine will use. These parameters include:

- The angle space between the rays that are launched from the transmitter in degrees
- The width of the Fresnel zone that is used to determine the partially obstructing edges.
- The maximum number of times the ray is allowed to reflect from the buildings.
- The maximum number of diffractions at the vertical edges of the buildings that are allowed.

These parameters can be set differently for each transmitter and allows the engine to perform a different simulation for each transmitter.
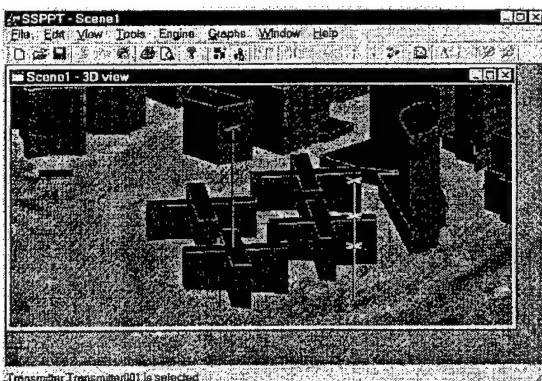

The following information is stored for the transmitter: transmitter name, X, Y, Z coordinates and the array of heights. Transmitter properties can be modified by double-clicking on the desired transmitter in the 3D perspective view or in the transmitter list view. The transmitter is checked to

Figure 23: View of the transmitters.

ensure that it has unique name and at least one height is specified. A check is also performed to determine that all transmitter heights are unique. Transmitters can be selected, moved and deleted and all these operations can be undone/redone. A transmitter changes color from green to yellow when selected. The selected transmitter can be moved horizontally in the 2D top view and vertically in the 3D perspective view. Figure 23 shows an example of two transmitters, one is located on top of roof and has only one height specified. The other transmitter has three heights specified and is located at ground level.

E. Displaying the Vegetation

Vegetation is currently loaded from ASCII file similar to the format of the building data file. The information that is 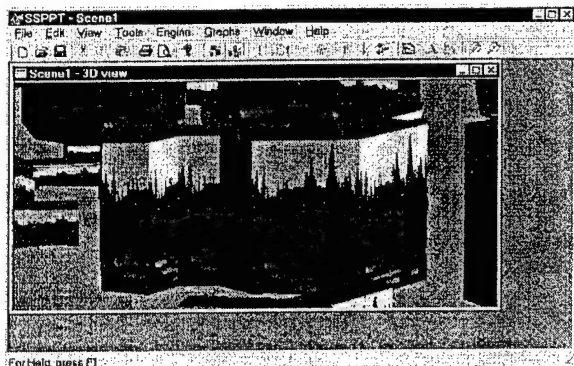store for the vegetation include: the foliage area number, the X, Y and Z coordinates of the vertices that make up the polygon that encloses the area, the average height of the vegetation within the area and the type of vegetation (i.e. dielectric constant). Convex and non-convex polygons can be used to specify the foliage. No move or delete operations are currently supported, they'll be implemented later. Foliage is

Figure 24: Viewing vegetation in the scene.

distinguished from buildings by using a solid green color for the top and an appropriate texture for the sides as shown in Figure 24. The model of the vegetation is intended to represent a modestly large area of vegetation and not a single tree.

F. Image Viewer

The Application also has built-in image viewer for images in the bitmap (.BMP) and JPEG (.JPG) formats. The image viewer is used to compare the virtual model of the physical environment with the true scene. Figure 25 shows an example of the aerial photo for a small area and is compared to the modeled buildings and terrain. It is often convenient to have the ability to view the image of the physical environment since it

allows the user to check the accuracy of the database without the need for an onsite survey. The viewer allows the image to be scrolled and zoomed so that more detail can
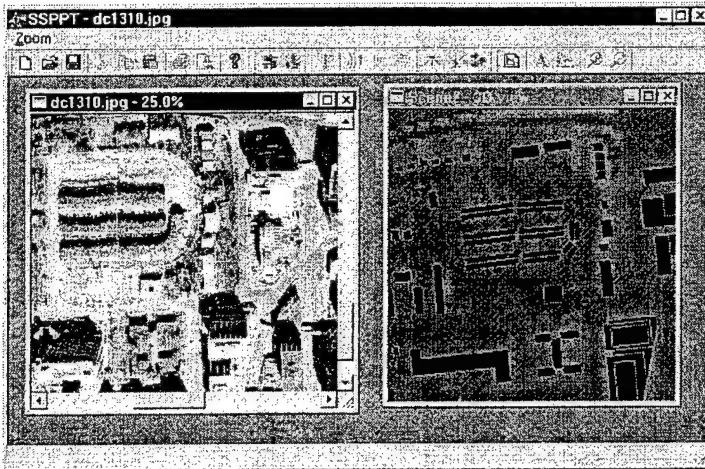

be viewed. The details that can be distinguished are dependent of the resolution of the image. Future versions of the application will allow a larger number of different image formats (GIF, Postscript, Sun Raster and etc) to be viewed.

Figure 25: Viewing a JPEG image.

G. Launching a Simulation

The minimum objects that are needed to order to conduct a simulation are at least one transmitter, one receiver and the buildings or the terrain data. A computational engine is selected from the Engine menu item. Currently only the VPL engine has been
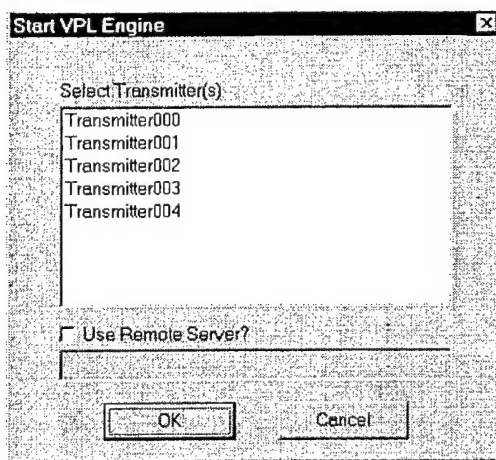

Figure 26: Transmitter selection dialog of the VPL engine.

integrated into the application. A dialog will appear that will list all the transmitters that have been place into the scene as shown in Figure 26. The user must select at least one transmitter to use in the simulation. Additional transmitters can be included in the simulation by simultaneously holding down the Ctrl key while selecting the additional transmitters from the list. The parameters that will be used in the simulation are unique to each transmitter and is specified in the transmitter properties dialog

shown in Figure 20. The number of reflections specified dictates the number of reflections between each segment of diffraction that the ray will undergo before terminating. Therefore, if one diffraction is used then the engine can potentially find a

ray that has undergone twelve reflections, six before and six after the vertical edge diffraction. The number of diffractions is for diffraction around the vertical edge of the building and must be between 0 and 2. The program will display an error message if there are no receivers or transmitter in the scene when a computational engine is selected.

When the transmitter(s) are selected the user starts the simulation by clicking the OK button. A progress dialog will appear to display the progress of the simulation and messages generated from the engine as shown in Figure 27. The progress dialog also contains a Cancel button that enables the u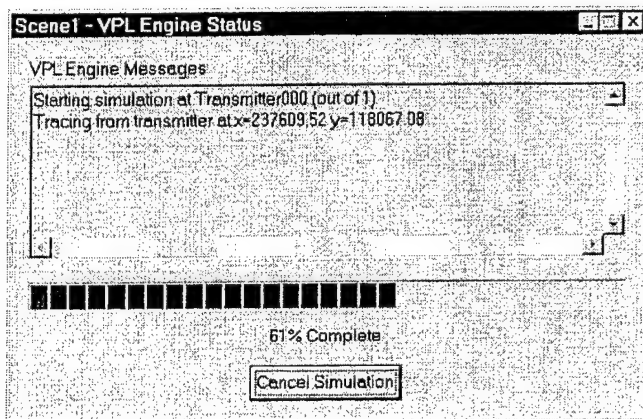ser to terminate the simulation at any time. The results of the simulation are stored as individual ray associated with each receiver. The information stored for each ray includes: the total path length, the received power, the departure angle from the transmitter and the arrival angle to the receiver (in azimuth and elevation). The 3D coordinate of the each segment of the ray path is also stored so that a graphical display of the ray can be made.

Figure 27:Progress dialog for the VPL engine.

H. Displaying the Simulation Results

The propagation prediction engine calculates the strongest powers of the rays that travel from the transmitter to the receiver above a preset minimum. To accomplish this, the engine determines the interaction of the ray with the physical environment and calculates the associated losses. The engine can generate a large amount of detailed information and viewing this data in raw tabular format would not provide a convenient way of understanding the results. To make the data come alive, SSPPT has and will incorporate many different ways to display the data. The application not only allows the user a great flexibility in generating full color charts and graphs, but it will also enable the results to be visualized with the 3D scene the physical environment and the rendered databases.

There are currently three ways to display the results of the SSPPT engine:

1. Receiver discs.
2. Ray path display.
3. Line plots and graphs.

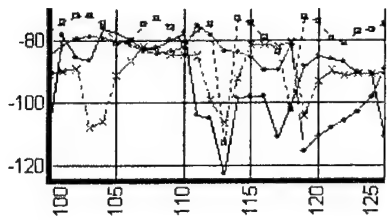Figure 29 shows an example of receiver discs displayed with the buildings and terrain. Receiver discs are color circles locat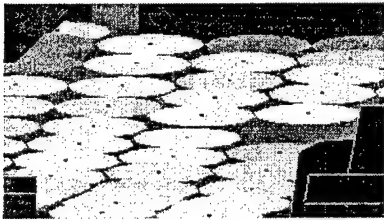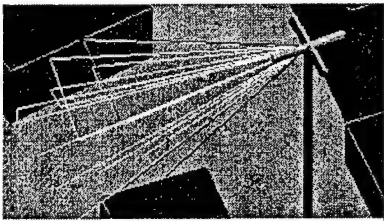ed around the receivers and are intended to display the distribution of the results over an area. The color and brightness of the disc under the receiver indicate some aspect of the data for that receiver that you have asked the system to display. The ray path display shows the paths that rays travel from the transmitter to the receiver(s) as shown in Figure 30. Like the receiver discs, the color and brightness indicate some value of the ray that you have chosen to display. The line plots are standard paper charts and graphs that allow the user to plot different variables of the data against each other on X and Y-axis as shown in Figure 28. This portion of the program allows the user great flexibility in the presentation of the graphs, perfect for comparing different variables of the results and for making reports and presentations.

Figure 28: Line graph of results.

Figure 29: Receiver discs.

Figure 30: Ray path display.

H.1 Receiver Discs

Receiver discs provide a way to view the simulation data for the receivers that is very easy to comprehend. The value of interest for that receiver is displayed as a colored disc around the receiver. As a result it is easy to see how receivers' placement around buildings and terrain effect how it receives the signal. In order to display data about the receivers as discs it is necessary to run
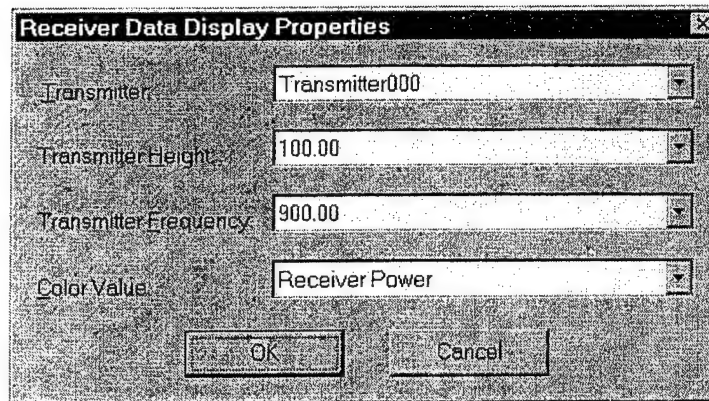

Figure 32: Receiver discs data selection dialog.

the engine to obtain the results. The simulation must include one or more receiver and at least one transmitter. Including the building and terrain database is optional. The depressed button shown in Figure 32 will bring up a dialog. The dialog shown in Figure
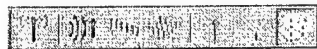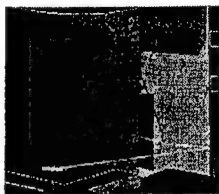

Figure 31: Receiver discs output button.

31 allows the user to choose the data belonging to a specific transmitter location, at one height (if the transmitter has more than one height specified) and the operating frequency. The bottom combo box provides a list of different values that can be computed for each receiver and displayed as the color of the disc. The application currently comes with a certain number of color value functions predefined; in the future users will be allowed to add custom functions.

H.2 Ray Path Display

Displaying the ray path within the scene is a powerful tool for understanding how signals propagate through the physical environment. When rays are displayed the user can graphically see the paths that the signal will travel from a transmitter to a receiver. For example, the weak signal for a receiver behind a building is because the signal diffracts

over the top of a tall building, while another receiver's signal is strong because it reflects from a nearby building without undergoing any diffractions.

The ray path information is only available after the simulation engine has started. The simulation must contain one or more receiver and at least one transmitter. The building and the terrain data are optional. Rays are displayed going from any number of transmitters to any number of receivers. There are two ways to choose which transmitters and receivers display their ray data. If the location of your transmitter or receiver is known then the user can select the appropriate button on the toolbar and then click the receiver or transmitter in the scene. Otherwise, if the name or ID number of the transmitter or receiver is known, the user can select the name or ID from lists in the ray properties dialog.

There are two buttons that can be selected to initiate the displaying of the ray paths. When the button shown in Figure 33 is depressed (the button showing a transmitter with rays coming out),

Figure 33: Ray display from transmitter selection button.

and a transmitter is selected in the scene, the ray properties dialog will appear with the selected transmitter highlighted. When the button shown in Figure 34 is

Figure 34: Ray display to receiver selection button.

depressed (the button showing a receiver with rays coming in), and a receiver in the scene is selected, the ray properties dialog will appear with the selected receiver highlighted. Once the ray properties dialog is closed, rays from additional transmitters can be displayed by depressing the transmitter ray display button (Figure 33) and selecting another transmitter. The ray properties dialog appears so that the user can choose the rays for a single height and single frequency. Additional paths can appear to more receivers by clicking the receiver ray display button (Figure 34) and then selecting a new receiver.

A set of rays can be undisplayed by choosing an appropriate ray display button, and then selecting a receiver or transmitter that is displaying rays. All the rays that arrive at the selected receiver or depart from the selected transmitter will be undisplayed. Rays

can also be removed from the scene by selecting the transmitter's name or the receiver's ID from the lists in the ray properties dialog.

When a ray display button is depressed for the first time and a transmitter or receiver is selected, the ray properties dialog shown in Figure 35 will appear. The bottom "Receiver Points" list shows all the recei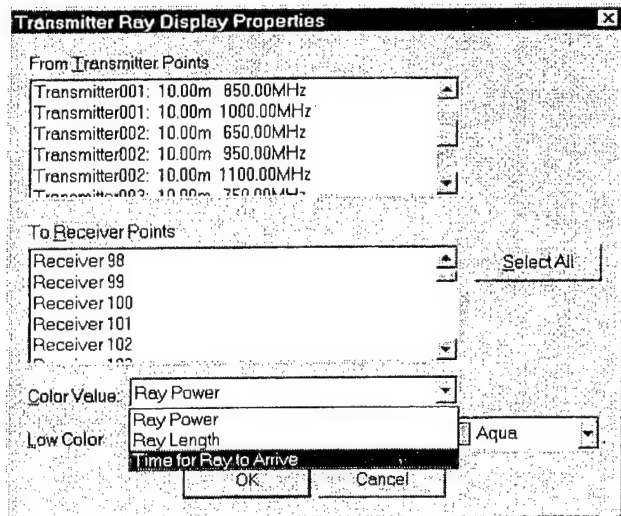vers in the scene regardless of whether they are displaying rays. Only the receivers whose ID number is highlighted will display the rays that are arriving into it. To highlight more than one receiver, hold down the **Ctrl** key while selecting receivers. To highlight a whole range of receivers highlight the first receiver in the range, and then hold down the **Shift** key while highlighting the last receiver in the range. The "Color Value" combination box allows you to select a piece of information about the ray to display as its color. These calculated values include:

Figure 35: Ray properties dialog.

- Delay Spread
- Angle Spread
- Total Power
- Direct Distance
- Cumulative Probability of Result

Different levels of brightness or different colors will be given to each ray based on how high or low the calculated value is.

The top "Transmitter Points" list shows a transmitter name, a height on that transmitter, and a frequency of that transmitter for all of the transmitters that will be displaying rays. A single transmitter may appear more than once if that transmitter will display rays from more than one height or frequency. All items in the list will display

rays, whether or not it is highlighted. To add a transmitter to the list, hit the add button and the dialog will changed to the one shown in Figure 36. In the bottom half of the dialog three combination boxes will appear that allow the user to choose a different transmitter name, height, and frequency. When all the selections have been made, the transmitter is added as a line in the top list box when the **OK** or **Add** button is depressed. To remove a transmitter, highlight its info in the list and hit the **Remove** button. Hit **Cancel** or **Remove** to change the

Figure 36: Add transmitter dialog.

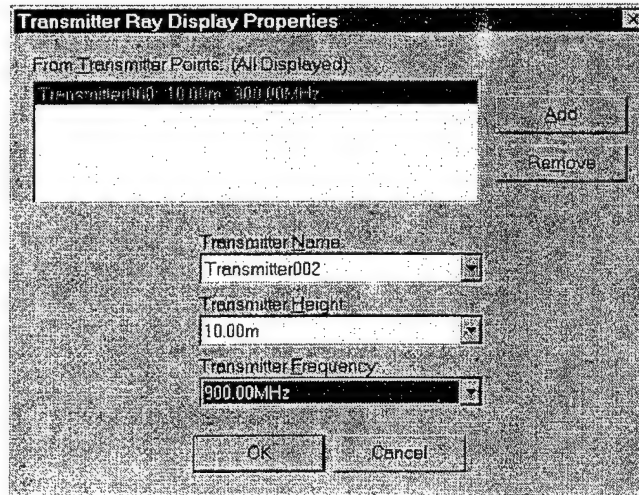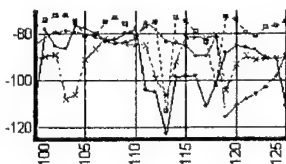dialog back to its normal appearance without adding a line to the list of transmitter points.

The dialog in Figure 36 will also appear when the transmitter ray display button is depressed and a transmitter is selected. The transmitter name will already have the name of the selected transmitter showing and it will only be necessary to select the height and frequency before continuing. At least one transmitter point must be listed, and at least one receiver selected before the rays are displayed in the scene. The **Cancel** button can be hit at any time to leave the dialog without changing how rays were displayed before the dialog was opened.

H.3 Line Graphs

Graphs are an important tool for the analysis and presentation of the results. The application has a powerful built-in line-graphing tool that allows great ease and flexibility in the creation of plots and graphs and in controlling their appearance. The SSPPT line graph tool is a what-you-see-is-what-you-get (WYSIWYG) editor that ensures the graph will appear the same as it is displayed when it is printed.

To display data in a graph it is necessary to start a simulation session with the engine. The simulation must contain one or more receiver and at least one transmitter.

Figure 40: Line graph button.

Including the buildings and terrain is optional. To open a new graph, click on the button shown depressed in Figure 40 or select New Graph from the Graphs menu. A blank window like the one shown in Figure 29 will appear. Each graph window represents one printed page. The blank white portion represent the page and all elements of the graph must lie inside this area. If the window is smaller than the size of the page then scroll bars will allow the user to move to different areas of the page. To see more or less of the page at once, the graph can zoom in or out. To

Figure 39: Blank graph view page.

Figure 38: Zoom in button.

zoom in depress the button shown in Figure 38 or through the zoom in item in the Graphs menu. To zoom out depress the

Figure 37: Zoom out button.

button shown in Figure 37 or selecting the desired zoom percentage from the Graphs menu.

The Print Setup dialog (Print Setup from the File Menu) shown in Figure 41 allow for the customizing of the graph page. The paper size, portrait or landscape orientation, and margin sizes can be set from this dialog. The changes made in this dialog are reflected in the size of the white page area in the graph window. The graph window and the physical environment scene keeps their print setup information separate, so that a graph can be printed portrait while the scene is printed landscape without interfering with each other. Future versions of the application will have a page control added to the bottom of the graph window so that all the graphs are in
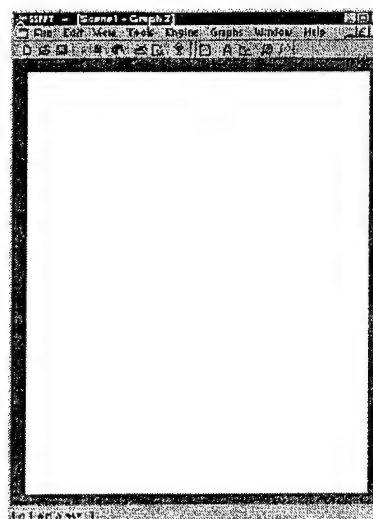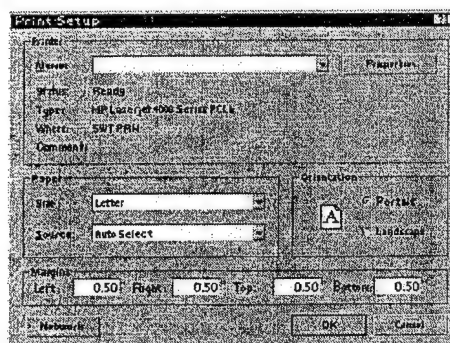

Figure 41: Print setup dialog.

one window rather than having one window for each page. This will also allow the user to print in order all pages at once.

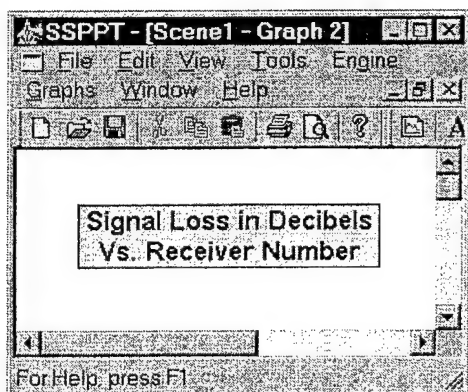When the size and orientation of the page has been determined it is then necessary to fill the page. A page can contain as much information as the user chooses. Multiple graphs, titles, charts, or any other available components can be added to a single page.

Figure 42: Text box for the graph view.

The simplest graph component is a text box shown in Figure 42. This component can be used as a title, a graph axis labels, an explanatory text, or any other kind of text on the page. To add a text box click on the New Text button shown in Figure 43, or select the Add Element/Text from the Graphs menu. Then hold the left mouse button down in the white page area where the upper right hand corner of the box of text is to be located and move the mouse to where you want the lower right hand corner of the text before letting go of the mouse button.

Figure 43: New text box button.

After selecting the page area in which the text is to appear the text properties dialog shown in Figure 44 appears. The dialog allows the user to type in the text and to choose the appearance of the text within the text box. The text can have different horizontal and vertical alignments as well as different orientations such as vertical orientation where the text is rotated $90°$ from the horizontal. The color of the background and a printed border around the text area can be set. When the Set Font button is clicked
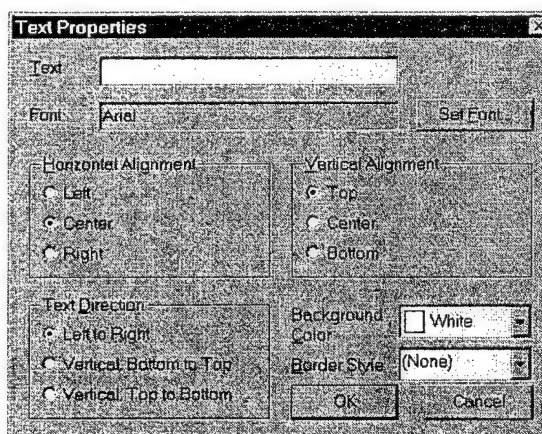

Figure 44: Text properties dialog.

a dialog will appear that allows the user to choose the font type, size, bold, italic, underline, and color to the actual text.

When the OK button is hit, the text will appear in the area you selected. The text box can be moved and re-sized by using the mouse to select the box from the graph view and dragging the mouse to the new position or size. When the cursor moves inside the text area, a dashed border with square

Signal Loss in Decibels Vs. Receiver Number

Figure 45: Cursor within a selected text box.

points on the corners and the center of each side will appear. When the cursor appears as a four-way arrow as shown in Figure 45, and by holding down the left mouse button and dragging the mouse will move the text area around the page without resizing it. When the cursor is positioned over a square point on the center of the side, the cursor will change into either a horizontal or vertical 2-way arrow, allowing the user to move that side and resize the area by dragging the point with your left mouse button. When the cursor is positioned over a corner, the cursor changes to diagonal 2-way arrow, allowing the user to adjust the size of the text box equally in width and height.

When the right mouse button is clicked inside the text area, a menu of other
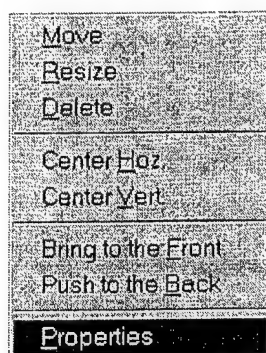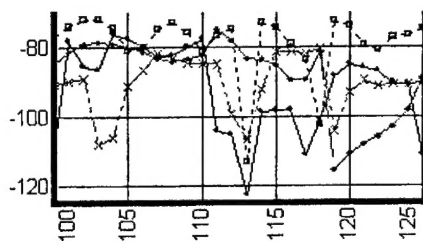
Move
Resize
Delete
Center Hoz.
Center Vert
Bring to the Front
Push to the Back
Properties

Figure 46: Text area menu items.

options shown in Figure 46 becomes available. The move command centers the text area on the next position that the mouse is click. The resize command moves the text area into a rectangle that is selected by dragging the left mouse button from the upper left corner to the lower right corner of the box that the text area is moved to. There are menu items that automatically center the text area horizontally or vertically in the page. The "Bring to the Front" and "Push to the Back" control whether the text area will show up over any other graph element it overlaps with. Selecting Properties brings back the dialog that appeared when you first created the text area, allowing the user to make changes to the text or appearance. The directions for moving and making changes to the text area also applies to all other graph elements.

The line graph tool provides amazing power to view the results from the SSPPT engine in a customized line graph. A new line graph can be added by hitting the New Line Graph button shown in Figure 47 or by selecting the Add Element/Line Graphs from the Graphs Menu. The new graph must then be positioned in the graph view

Figure 47: Line graph button.

by dragging the mouse cursor over the area you want the graph to fill. The graph properties dialog has a tab control at the top that allows the user to access three pages that control different aspects of the appearance of the graph. The first tab is appearance and controls the color and different types of graph lines. The appearance page also has a button that will add a legend to the graph page. The legend can be moved, resized and customized like any other graph component. The data page controls the scale of the axis and the interval spacing of the minor and major grid lines. At the bottom of the data page is the list of data sets that are graphed. Highlighting a data set and clicking properties brings up more information about what data will be graphed, and how it will appear on the graph. The labels page, the last tab on the dialog, controls the appearance of the tick values of the major and minor axis.

The Appearance page controls the type of lines and the color of the axis and grid of the graph. Figure 48 shows the layout of the appearance page of the axis properties dialog with its combo boxes and buttons. There are combo boxes to chose the background color of the graph, the color that the axis and style of grid lines that will be displayed. The grid can include major and minor ticks on each axis, as well as the origin and axis lines, and each can be specified with different widths of solid line, different types of broken lines, or to display no grid lines at all. The generate legend button automatically creates and places a special text box on the page that lists each data set name next to the data set's line style.
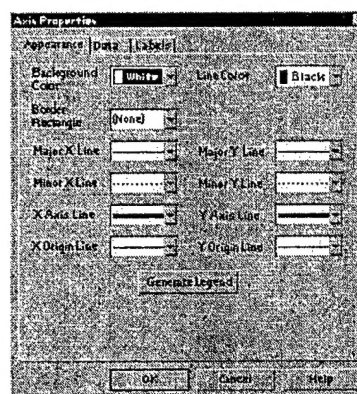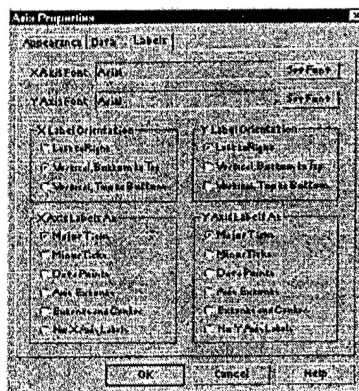

Figure 48: Axis properties dialog.

The Data page of the Axis properties dialog shown in Figure 49, allows the user to set the minimum and maximum plotted value for both the X and Y axis, as well as the major and minor tick intervals (which are used to decide where to draw the graph's grid lines). The Auto Scale button automatically calculates the minimum and maximum by find the range from the data sets to be graphed and sets these numbers so that the graph will be large enough to plot all the data. The Auto Scale button will also sets the tick intervals to display a reasonable number of grid lines. The units' multipliers control the numbers that are automatically printed on each axis. For example, if the data is in billions, by specifying that each number should be multiplied by 0.000000001 it will prevent nine zeros being displayed after each number.

Figure 49: Data properties for the axis

The most important part of the dialog is the list of data sets at the bottom of the dialog. This list shows the data lines that will be drawn into the graph. Additional data sets can be added or remove them at any time. It is a good idea to hit auto scale anytime a new set of data is added to the list. When the Add Data Set button is pressed or when a data set is highlighted in the list and the Properties button is pressed, the Data Set Properties dialog will appear. In the Data Set Properties dialog is where the user can control what data is graphed, what the data points like, and what the lines connecting those points look like.

Figure 50: Axis label dialog.

The Labels page shown in Figure 50 controls how the numbers on the axis are displayed. Hitting the Set Font button and changing the settings on the font dialog can change the font and font size of the numbers. The radio buttons in the dialog allows the user to customize the alignment of the numbers so that they are horizontally or vertically positioned. The buttons also sets up how often a

46

number is printed.

The Data Set Properties Dialog shown in Figure 51 controls what data is graphed as well as how it will appear displayed.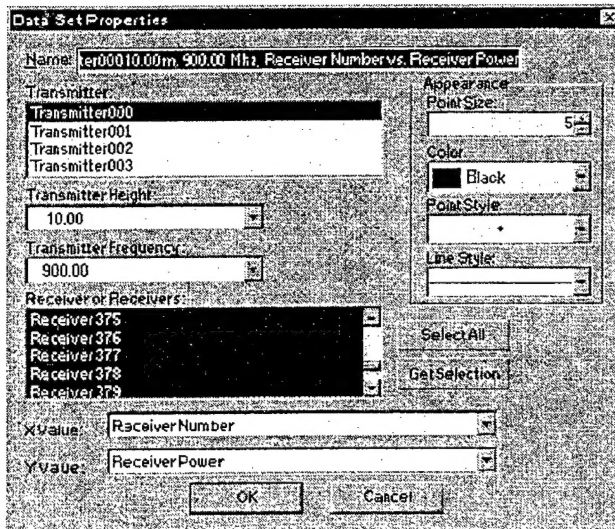 To graph data, the result from a specific transmitter at a specific height and frequency must be selected from the list. In addition to the transmitter a receiver or group of receivers must also be selected. The two combo boxes at the bottom allow the user to choose which facet of the data is extracted to use as the X and Y values. While only a few options are available now, this list will be expanded to include many more types of data and an option for custom calculations is being considered. The Appearance section of the dialog allows the user to control how the data will be graphed. Various types and sizes of points, different sizes of solid lines, different types of broken lines, and even no lines at all when a scatter plot is desire can be selected. Different colors can also be selected for each set of results in the graph to enhance the differentiation of the data.

Figure 51: Data set properties dialog.

In future version of the application, this dialog will be expanded to enable data from other sources such as spreadsheets, databases, and text files to be imported and plotted along side the SSPPT generated data. This feature will allow the user to graph predicted versus measured data or to compare the SSPPT data against predictions made with other algorithms. There will also be more types of page elements in the future. Different types of graphs, such as pie chart, bar graph, and 3D graph will be included. Other features might include the ability to add graphics, tables, and perhaps even Microsoft Office documents. The ability to save the graph pages as a file or as an image

as well as the ability to export the data sets into different file formats such as: text file, spreadsheet, and database will also be implemented.

I. Proprietary File Format

In order to retrieve the set of simulation results that correspond to a physical environment it is necessary to store all elements of the scene and the results of the propagation engine. A proprietary binary file format is being developed that would include all the information about the scene (i.e. buildings, terrain, vegetation, transmitter and receiver locations) and the results from the simulation in one file. This would enable the user to conduct a series of simulations, save the results and retrieve them for analysis at a later time. Since the simulation results are permanently linked to a scene there would not be any confusion regarding whether the results correspond to a particular physical environment. Some of the advantages of storing the data in a binary format instead of a regular ASCII text file include a 60% decrease in disk space and improved loading speed.

J. Printing

Printing different elements of the view windows has been implemented for the 3D scene view and for the output line graphs. All aspects of the 3D scene view can be printed including the ray path and the receiver discs along with the building, terrain, vegetation, receivers and transmitter(s). Printing the line graphs is particularly useful since it is often necessary to include the plotted results in reports and presentations.

K. Grid

A feature that allows the user to select/unselect the display of a grid within the scene has been implemented. The grid can be turned on or off from the Display menu item. The grid lines will visually aid to judge approximate distance between the objects. Numbers are displayed along the edges to help to judge distance between the objects. The user is able to specify grid parameters such as grid interval and the number of minor ticks between the major ticks. As the scene is zoomed out the labels for the minor ticks are not shown if there is not enough space to display them without overlapping over labels.